

(19) 日本国特許庁 (JP)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平 10 - 187726

(43) 公開日 平成10年(1998)7月21日

(51) Int. Cl.⁶G 0 6 F 17/27
17/21

識別記号

F I

G 0 6 F 15/20 5 5 0 E
5 4 8 A

審査請求 未請求 請求項の数 6

O L

(全 19 頁)

(21) 出願番号 特願平 8 - 3 4 9 5 8 0

(22) 出願日 平成8年(1996)12月27日

(71) 出願人 000001007

キャノン株式会社

東京都大田区下丸子3丁目30番2号

(72) 発明者 三上 陵一

東京都大田区下丸子3丁目30番2号 キャノ
ン株式会社内

(74) 代理人 弁理士 谷 義一 (外1名)

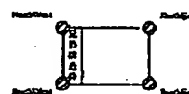
(54) 【発明の名称】 構造化文書編集システム

(57) 【要約】

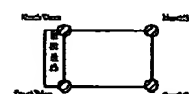
【課題】 SGMLのような汎用の構造化文書を取り扱う構造化エディタによって、表形式のレイアウトを作成・蓄積できるようにする。

【解決手段】 構造化枠の配置指定は大きく分けて、取り付け対象構造化枠の指定 (Attachment) と取り付け基準位置 (Attach Origin) の指定並びに基準位置からのずれ量 (Offset) からなる。取り付け対象構造化枠としては兄弟構造化枠または親構造化枠を選ぶことが可能である。兄弟枠を選んだ場合にはさらに兄弟枠の4つの端点の内のいずれかを取り付け位置に指定することができる (図9 (a))。一方、親構造化枠を選んだ場合は、構造化枠内側領域の4つの端点を取り付け位置に指定することができる (図9 (b))。図9

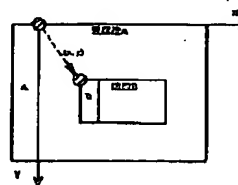
(c) には、構造化Bに関する配置指定の例を示した。この場合、親を取り付け構造化枠に指定し、取り付け基準位置は親のNorthWest、さらに取り付け指定位置からのずれ量として (x, y) を指定している。



(a)

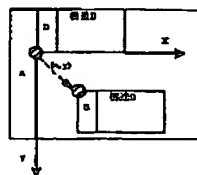


(b)



(c)

AttachPoint
AttachOrigin
AttachOffset
AttachType



(d)

AttachPoint
AttachOrigin
AttachOffset
AttachType

【特許請求の範囲】

【請求項1】 構造化文書内で表形式レイアウトを実現する構造化文書編集システムにおいて、各論理構造に対し相対的に表構造枠の配置を記憶する手段と、

前記記憶された配置に基づき表構造枠の配置を行う手段と、

前記配置された表構造枠の表示を行う手段とを備えることを特徴とする構造化文書編集システム。

【請求項2】 各論理構造に対し構造枠表示属性を記憶する手段と、

論理構造枠表示／非表示の表示モードを記憶する手段と、

前記構造枠表示属性と前記表示モードから構造枠を表示するか否かを決定する手段と、

前記決定された表示に基づき構造枠の表示を行う手段とを備えることを特徴とする請求項1記載の構造化文書編集システム。

【請求項3】 構造化文書内で表形式レイアウトを実現する構造化文書編集システムにおいて、

表の構造枠内の内容変更を検出する手段と、

現在挿入位置を検出する手段と、

表の構造枠内の内容領域を位置検出による大きさに変更する手段と、

編集対象とする表の構造枠の変更指示を検出する手段と、

表の構造枠のサイズを変更する手段と、

構造枠の配置を行う手段とを備えることを特徴とする構造化文書編集システム。

【請求項4】 構造化文書内で表形式レイアウトを実現する構造化文書編集システムにおいて、内容領域を変更後に枠内に内容領域をスクロールする手段を備えることを特徴とする請求項3記載の構造化文書編集システム。

【請求項5】 構造化文書内で表形式レイアウトを実現する構造化文書編集システムにおいて、表の構造枠内の内容領域を指定の大きさを変更する前に構造枠のサイズ及び配置を計算する手段を備えることを特徴とする請求項3又は4記載の構造化文書編集システム。

【請求項6】 構造化文書内で表形式レイアウトを実現する構造化文書編集システムにおいて、枠内の内容領域を指定領域をスクロールする手段を備えることを特徴とする請求項3記載の構造化文書編集システム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、文書処理装置等における構造化文書編集システムに関するものである。

【0002】

【従来の技術】SGML (Standard Generalized Markup Language、ISO 8879-1986) に規定されているような構造化文書は、文書構造を表す「論理構造要素」と文

書内容を表す「内容要素」から構成される。文書構造の一例を図1(a)に示す。図1(b)は図1(a)の構造を持った構造化文書の書式の例であるが、論理構造を表すためのタグと呼ばれる記号 (<...> ... </...>)と、文字列などの文書内容から構成されていることがわかる。

【0003】このような構造化文書を、構造化エディタでは、「文書型定義」と呼ばれるものにより、文書内で使用する論理構造を定めることができる。

10 【0004】構造化エディタは、構造化文書の編集開始に先だって、必ずこの文書型定義を読み込み、定められた文書型から逸脱しないように検証を行いながら、ユーザに文書編集を行わせる。

【0005】これにより、学術論文、操作手引き書、会計報告書などのような、多数の著者が介在しながら、文書に対し一定の構造を持たせる必要がある文書についての編集を行うことを可能としている。

20 【0006】また、文書型定義も構造化文書自身もテキスト形式で表現されることが定められており、異なる処理装置間で文書交換可能となっている。

【0007】図2は図1の構造化文書を編集するための文書型定義の一例を示したものである。

【0008】この例では、「報告書」構造内に「題目」「ヘッダ」、「章」構造要素がこの順番で現れ、なおかつ「章」構造要素については0回以上繰り返して出現しなければならないことが示されている（即ち、「章」構造は存在しなくともよい）。

30 【0009】「題目」構造内では文字要素（# PCDATA）が0回以上現れなければならない。「ヘッダ」構造内では「年」、「月」、「著者」、「概要」構造要素がこの順で現れ、かつ「著者」構造要素は1回以上繰り返して出現しなければならない。

【0010】「年」、「月」、「著者」構造内には文字要素、「概要」構造内には「段落」構造要素、「段落」構造内には文字要素のみが存在することが示されている（以下は同様なので省略する）。

40 【0011】ここで示した文書型定義は単なる一例であり、構造化文書内で扱う論理構造名や各要素の出現規則は、文書作成の目的に応じて個別に決められる性質のものである。

【0012】言い換えると、論理構造名自身は何ら体裁上の特別の意味を持たないものである。

【0013】例えば「段落」という構造名は、改行と字下げを伴ういわゆる段落レイアウトによって表示されることを何ら意味していない。

【0014】「段落」構造をいわゆる段落に見えるようにレイアウトするために、ユーザは構造化エディタに対し、「段落」構造を段落的にレイアウトするように別途指示を与えることを行う。

50 【0015】図3にユーザが各構造に対する体裁を指定

するためのレイアウト指示の例を示した。この例では、段落構造内の文字要素を表示する際のフォントの指定と文字開始行における字下げや改行の指示が行われている様子が示されている。構造化文書エディタは、構造化文書の編集を開始するに当たって、このようなスタイル指定を読み込んで画面表示を行うことができる。

【0016】構造化文書にスタイル付けを行い、画面上に表示する方式としては図4に示すようなものがある。図4(a)は論理構造を示しながら編集を行わせるための「タグ表示モード」といわれる方式であり、図4(b)は印刷出力などの最終レイアウト形式が示される「レイアウトモード(タグ非表示モード)」と云われる方式である。

【0017】さて、このような構造化エディタにおいて表を取り扱う必要が生じたとする。先に述べたように、SGMLなどの構造化文書には表形式にレイアウトを行うことが暗黙に定められた論理構造は本来存在しない。表を表すことを意図して導入した論理構造に対して罫線等を施して表形式レイアウトを行うためには、別途スタイルによる指定を行うべきであるが、従来のスタイル指定方式に専ら構造内に現れる文字などの内容要素の表示方式に関する指定のみしか行うことができず、表形式でレイアウトするための罫線などの扱いについて指定することはできなかった。

【0018】従来の構造化エディタには、表形式レイアウトを行えるものもある。しかし、それらにおいては、罫線や枠付けをスタイルで指定するのではなく、その構造化エディタのみが表と定めた特別な論理構造名を使用するものであった。

【0019】これは、異なる処理装置間での文書交換を前提とする構造化文書の処理方式としては優れた方式とはいえ、構造化文書の可搬性を損なう方式であるといえる。

【0020】図5(a)に従来の構造化エディタにおける表専用の論理構造の一例を示した。図5(b)は図5(a)の構造を表構造と認識する構造化エディタによって画面表示を行った例である。図5(b)には最終レイアウト形式を示すレイアウトモードにおいて表形式レイアウトが行われている様子が示されている。

【0021】この方式の場合には、構造化エディタ自体がTBLHEAD、TBLROWS、TBLROW、TBLCELLという論理構造を表と見做すと定めており、これらの構造名を認識すると自動的に図5(b)のようなレイアウトを行なうわけである。

【0022】

【発明が解決しようとする課題】以上述べたように、従来の方式においては、個々の構造化文書処理装置が、表形式のレイアウトを実現するため独自の論理構造を使用しているため、文書を作成した文書処理装置以外においては表形式レイアウトで表示・印刷できないという欠点があ

った。

【0023】本発明は、このような状況を鑑み、SGMLのような汎用の構造化文書を取り扱う構造化エディタによって、表形式のレイアウトを作成・蓄積できるようにすることを目的としている。

【0024】また、構造枠内に内容を入力するとき、構造枠内の入力された内容のレイアウト処理に続いて構造枠の再配置を行う場合には、枠の再配置計算とその結果の表示処理が頻繁に発生するため、ユーザが計算及び表示の処理待ちになるという欠点があった。

【0025】さらに、構造枠のサイズの変更に応じて編集集中の構造枠の位置が変更されるので、ユーザは、改めて編集集中の枠の位置と現在の挿入位置の確認を行わなければならない、ユーザの連続的な入力作業が妨げられるという欠点があった。

【0026】

【課題を解決するための手段】請求項1記載の発明は、構造化文書内で表形式レイアウトを実現する構造化文書編集システムにおいて、各論理構造に対し相対的に表構造枠の配置を記憶する手段と、前記記憶された配置に基づき表構造枠の配置を行う手段と、前記配置された表構造枠の表示を行う手段とを備えることを特徴とする構造化文書編集システムである。

【0027】請求項3記載の発明は、構造化文書内で表形式レイアウトを実現する構造化文書編集システムにおいて、表の構造枠内の内容変更を検出する手段と、現在挿入位置を検出する手段と、表の構造枠内の内容領域を指定の大きさに変更する手段と、編集対象とする表の構造枠の変更指示を検出する手段と、表の構造枠のサイズを変更する手段と、構造枠の配置を行う手段とを備えることを特徴とする構造化文書編集システムである。

【0028】本発明を用いることにより、独自の構造を使用することなく表形式レイアウトの実現が可能となった。このことにより、一般的な構造化エディタで作成・蓄積された文書を共有するような場合においても、表形式レイアウトで表示・印刷することが可能となった。

【0029】また、構造枠内の編集を行っている最中に、構造枠サイズや位置の変更を行わなくてもすむ。

【0030】

【発明の実施の形態】次に、本発明について図面を参照して詳細に説明する。

【0031】図6は、本発明の構造化文書編集システムのシステム構成の一例を示すブロック図である。

【0032】1は描画内容を表示するCRTであり、2はCRT1の画面に表示されるイメージを展開するビデオRAM(以下、VRAM)であり、3はメモリ間あるいはメモリと各デバイス間のデータ転送をコントロールするビット処理ユニット(以下、BMU)であり、4は処理を指定するための各種キーを備えたキーボードであり、5は画面上のアイコンなどを指し示すために用

いられるポインティングデバイスであり、6は本装置処理全体の制御を司るCPUであり、7はCPU6の動作処理手順を記憶しているROMである。このROM7には文書処理に係るアプリケーションプログラムやエラー処理プログラムをはじめ、本方式に係るプログラムも記憶されている。8は上述した各種プログラムをCPU5が実行する時に作業領域、一時退避領域として用いられるRAMを示している。9は、ハードディスクドライブ（以下、HDD）、10はフロッピーディスクドライブ（以下、FDD）をそれぞれ示し、11はネットワークインターフェース（以下、Net-I/F）を示す。12は上述した各ユニット間を接続するI/Oバス（アドレスバス、データバス及び制御バスからなる）である。

【0033】なお、本発明は、11を介して接続された複数の機器から構成されるシステムに適用しても1つの機器から構成される機器に適用してもよい。また、本発明は、ROM7以外にもシステム或いは装置にプログラムをHDD9やFDD10により供給することによって達成される場合にも適用でき、また本発明はシステム或いは装置にプログラムを供給するHDD9やFDD10等の記憶媒体にも適用できることはいうまでもない。

【0034】〔実施形態例1〕次に、図7を参照して、実施形態例1の動作を説明する。

【0035】図7は、図5に示した構造化文書における論理構造の親子関係を示した図である。

【0036】構造は下へ向かうほどより深い構造であることを示しており、横並びの並列構造については右へ向かうほどより文書内で後から出現することを示している。

【0037】図7に重ねられている矢印線は、文書に対し構造枠表示モードを与えた時の表形式レイアウトの処理順序を示している。構造枠表示モードには「文書全体について構造枠レイアウトを行う」、「文書全体について構造枠レイアウトを行わない」の2種類がある。

【0038】文書に与えられた構造枠表示モードは、子構造である段落へと通知され、段落の子構造であるTBLHEAD、続いてTBLROWSに通知され、この通知と共にレイアウト処理が実行される。TBLROWSに通知された表示モードは、最も最初に出現する子構造であるTBLROW（最左端）に通知され、またさらに子構造であるTBLCELLに伝達されながらレイアウト処理が行われる。全てのTBLCELLにTBLCELLの出現順に通知した後、残りのTBLROWへと通知が行われる。

【0039】以上のように表示モードの通知並びにレイアウト処理は、並列構造よりも下位構造に対し優先的に行われる。

【0040】これは、ある子構造に対してレイアウト要求を行った場合、それが終了した時点でその全ての子孫構造に渡ってレイアウト処理が完了することを示すもの

である。

【0041】図8に各論理構造に対して与えられる構造枠表示属性の例を示した。

【0042】この例では、文書と段落に対して「構造枠表示モードに従う」属性が与えられている。即ち、これらの構造は構造枠を表示する旨のモードが通知された場合には構造枠を表示し、構造枠を表示しない旨のモードが通知された場合には構造枠を表示しないように振る舞う。同時に、構造枠表示時には「構造名を表示する」よう指示が与えられている。

【0043】一方、他の構造は全て必ず「構造枠を表示する」という属性が与えられている。この場合には、構造枠表示あるいは非表示のいずれかの指示が通知された場合でも必ず構造枠を表示するように振る舞う。同時にこれらの構造枠に関しては「構造名を表示しない」よう指示が与えられている。

【0044】続いて図9、図10に構造枠の配置指定の例を示す。

【0045】図9は、構造枠の配置の指定方式の考え方について説明したものである。構造枠の配置指定は大きく分けて、取り付け対象構造枠の指定（Attachment）と取り付け基準位置（Attach Origin）の指定並びに基準位置からのずれ量（Offset）からなる。

【0046】取り付け対象構造枠としては兄弟構造枠または親構造枠を選ぶことが可能である。兄弟枠を選んだ場合にはさらに兄弟枠の4つの端点の内のいずれかを取り付け位置に指定することができる（図9（a））。一方、親構造枠を選んだ場合は、構造枠内側領域の4つの端点を取り付け位置に指定することができる（図9（b））。ここでは、親と兄弟における基準点の違いを明らかにするために構造名を表示する構造枠を用いて説明している。

【0047】図9（c）には、構造Bに関する配置指定の例を示した。この場合、親を取り付け構造枠に指定し、取り付け基準位置は親のNorthWest、さらに取り付け指定位置からのずらし量として（x，y）を指定している。

【0048】図9（d）では、構造Bに関する別の配置指定の例を示した。この場合は、兄弟構造Bを（自分自身も兄弟の一員である）取り付け指定構造枠とし、そのSouthWest端点を取り付け位置に指定する。さらに、ずらし量（x，y）も指定している。

【0049】実施形態例1では、取り付け指定構造名に合致する構造枠が複数存在する場合は、最も直前に出現した構造枠が取り付け対象となり、合致する構造枠が以前に1つも存在しない場合は親のNorthWest端点に取り付けられるものと定めている（後に出現する弟構造は名称が合致していても取り付け対象とはしていない）。

【0050】図10は、図7に示した文書構造に対し、構造枠の配置を指定した例を示している。

【0051】図10の1～8行では、TBLHEADの子供としてTBLROWSが現れた場合にTBLROWSがどのように配置されるかを示している。TBLROWSは親構造であるTBLHEADのNorthWest端点に取りつけられるように指示されている。これが表示された場合の様子は図11(a)に示されている。

【0052】図10の10～17行では、TBLROWSの子供としてTBLROWが現れた場合のTBLROWの配置について指定が行われている。TBLROWは直前のTBLROWのSouthWest端点に接続されるが、TBLROWS中で最も最初に現れるTBLROWの場合、直前にTBLROWが存在しないため、親構造であるTBLROWSのNorthWest端点に取りつけられる。この指定により、複数のTBLROWは重なることなく順々に縦方向に並んでいくことになる。これが表示された場合の様子は図11bに示されている。

【0053】図10の19～26行にはTBLROW中のTBLCELLの配置の指定例が示されている。この場合は、取り付け指定位置がNorthEastであるため、TBLCELLは重なることなく順々に横方向に並んでいくことになる。これが表示された場合の様子は図11cに示されている。

【0054】図7から図11に渡って説明した内容を組み合わせると、「文書」構造に対し構造枠表示指示を与える場合、図8の属性に基づき、下位の全ての構造が構造枠を表示し、さらにTBLHEAD以下の構造の場合、図10の配置規則に従い表レイアウトを行うことができる。それを示したのが図12(a)である。これで分かるように、文書、段落の構造枠が表示されている。

【0055】逆に、「文書」構造に対し、構造枠非表示指示を与えた場合、TBLHEAD構造より下位の構造のみが構造枠となり、その配置規制に従って表レイアウトを行うことができる。それを示したのが図12bである。

【0056】図13～図15に実施形態例1の方式の動作手順を表すフローチャートを示した。図13において、まず、各構造に対し構造枠表示属性を与え(ステップS301)、各構造に対し構造枠の配置の指定を行う(ステップS302)。

【0057】ここにおける指定は必ずしも先に述べたような表示属性指定言語あるいは配置指定言語により行われるとは限らず、ダイアログ等対話形式で指定が行われる場合もあることは勿論のことである。

【0058】次に、ユーザによって表示モードの指定が行われるかどうかを判定し(S303)、表示モードの指定が検出されなかった場合にはステップS303へと戻る。

【0059】一方、表示モードの指定が検出された場合には、最上位構造を得て(ステップS304)、それに対して表示モードを与えてレイアウト要求を行う(ステップS305)。このレイアウト処理の詳細については、後で図14を用いて詳しく説明する。

【0060】レイアウト処理が終了したら、表示を行う必要のある範囲を得て(ステップS306)、最上位構造に対し表示範囲を与えて表示要求を行う(ステップS307)。

10 【0061】ここで、表示を行う必要範囲は例えば、文書がスクロールウィンドウ内に表示されている場合にはウィンドウの可視領域などが与えられる。

【0062】この表示処理の詳細についても、後で図15を用いて詳しく説明を行う。

【0063】続いて、図14を用いて、図13のステップS305の処理の詳細について説明を行う。

【0064】まず、表示モードを与えられてレイアウト処理を要求された構造は、自構造の構造枠表示属性を得る(ステップS401)。続いて、表示属性が「必ず構造枠を表示する」であるかどうかの判定を行う(ステップS402)。もし、必ず構造枠を表示するのであれば、構造枠を作成する(ステップS404)。一方、必ず構造枠を作成するでない場合には、「構造枠表示モードに従う」であるかどうかを判定し(ステップS403)、表示モードに従う場合には、ステップS404へ進む。逆に、表示モードに従わない場合には(絶対構造枠を表示しない指定であるので)構造枠作成ステップを省略してステップS407へと進む。

30 【0065】ステップS404において構造枠が作成されたら、次に構造名を構造枠内に表示するかどうかを判定する(ステップS405)。構造名を表示する場合には、構造名を構造枠に取りつける(ステップS406)。構造名を表示しない場合には、ステップS407へと進む。

【0066】ステップS407では、自構造内に文字要素が存在するかどうかを調べ、記憶する。これは、文字と構造枠が混在する場合には構造枠をあたかも1つの文字であるかのようにレイアウトするための処理である。

40 【0067】このステップは現時点で文字要素が全く存在しないかどうかの判定だけでなく、本構造内には将来的にも文字要素が入力されない(本構造が文字入力を許容しない構造である)かどうかを判定し、現在も将来も文字が入り得ないか否かを記憶するステップに変更してもよい。その場合には、ステップS413も同時に変更を要する。

50 【0068】次に、構造内の全ての子要素(文字あるいは構造)を出現順に得る(ステップS408)。何らかの子要素が得られたかの判定を行う(ステップS409)、要素が得られた場合にはさらにそれが構造要素であるかどうかの判定を行う(ステップS410)。

【0069】構造要素であった場合には、子構造に対し、先に自構造に対して与えられた表示モードを与えてレイアウト要求を行う（ステップS411）。

【0070】子構造に関するレイアウト要求が終了した時点で、その子構造が構造枠を表示するかどうかの判定を行う（ステップS412）。子構造が構造枠を表示すると判定された場合は、さらにステップS407において記憶した文字要素存在可能性の判定を行う（ステップS413）。

【0071】構造内に文字要素が存在しない場合は、さらに自構造が構造枠表示を行うかどうかの判定を行い（ステップS414）、構造枠を表示する場合には、ステップS408に戻って次の要素に関する処理を継続する。このように、ステップS412、ステップS413、ステップS414が同時に満たされる場合は、親子共々全ての要素が構造枠となる（もちろん文字要素は現れない）場合に等しい。

【0072】さて、ステップS410において、構造要素以外の要素が得られた場合は、得られた要素は文字要素であるので、文字としてのレイアウトを行う（ステップS415）。

【0073】ステップS413において構造内に文字要素が存在すると判定された場合と、ステップS414において自構造が構造枠表示を行わないと判定された場合には、構造枠の配置指定は適用されず、子構造枠があたかも1つの文字であるかのようにレイアウトが行われる（ステップS415）。

【0074】ステップS412において、子構造が構造枠表示を行わないと判定された場合にはすべてに子構造内でのレイアウトが終了しているので、そのままステップS408へと戻る。

【0075】このようにして次々に子要素のレイアウトを行い、最終子要素の処理が終わるとステップS409にて枠内要素が得られないとの判定が得られる。

【0076】すると、再び自構造が構造枠を表示するかの判定（ステップS416）と、子要素が全て構造枠であるかどうかの判定（ステップS417）が行われ、この2つの条件が同時に満たされる場合にのみ、子構造枠の配置指定を読み（ステップS418）、全ての子構造枠の配置を決定する（ステップS419）が実行され、最後に自構造枠のサイズを決定し、自構造内レイアウトを終了する。

【0077】ステップS416において自構造が構造枠を表示しないと判定された場合には、既にステップS415によって全ての子要素をあたかも文字要素であるかのごとくに配置済であるのでそのままレイアウト処理は終了する。

【0078】ステップS417において子要素に構造枠でないものが存在する（文字要素も含む）と判定された場合は、やはりステップS415における子要素のレイ

アウトの結果に基づいて自構造枠のサイズを決定し（ステップS420）、構造内レイアウトを終了する。

【0079】最後に図15を用いて、図13のステップS307の表示処理の詳細について説明を行う。

【0080】まず、表示要求と共に与えられた表示範囲を取得する（ステップS501）。次に、自構造は構造枠を表示するかどうかの判定を行い（ステップS502）、構造枠を表示する場合には構造枠の表示範囲と与えられた表示範囲との積領域を求め（ステップS503）、その積領域に関して構造枠部品の表示を行う（ステップS504）。もし、構造枠と同時に構造名も表示する指示が行われていた場合には、この構造枠部品は構造名も含む。また、積領域が存在しない場合には、表示を行う必要もないばかりか子構造に表示指示を伝達する必要もないことは言うまでもない。

【0081】一方、ステップS502において子構造が構造枠を表示しない場合には、構造枠に関する処理は一切行わず、ステップS515へと進む。

【0082】構造枠の表示処理が終了した時点で、構造内要素の表示を行う。

【0083】まず、構造内の子要素を出現順に得（ステップS505）、何らかの子要素が得られたかどうかの判定を行う（ステップS506）。

【0084】子要素が得られた場合には、さらに構造要素であるかどうかの判定を行い（ステップS507）、構造要素である場合には先にステップS503において算出した積領域を与えて表示要素を行う（ステップS508）。

【0085】この要求は再帰的に子孫構造へと伝達され、要求が終了した時点ではその全ての子要素の表示が終了している。一方、ステップS507において、構造要素でないと判定された場合、出現要素は文字要素である。この場合、文字要素の表示されるべき矩形領域が与えられた表示領域に含まれるかどうかを判定し（ステップS509）、一部でも含まれると判定された場合には文字要素を表示する（ステップS510）。

【0086】一子要素の処理が終わると再びステップS505に戻り次の子要素の処理を続ける。処理終了の結果、全ての子要素処理が終了した場合には、ステップS506にて子要素が得られない旨の判定が行われ、表示処理は終了する。

【0087】〔実施形態例2〕上記のように構造枠が設定された表形式で、構造枠中に文字や構造を入力していくと、当初設定した構造枠の大きさの中に、内容が入り切らなくなることが起きてくる。その様子を示したのが図16（a）である。この場合、図16（b）に示すように、構造枠内のレイアウト処理と同時に構造枠のサイズを調整した。そして、それとともに変化した構造枠が他の構造枠と重ならないように構造枠の再配置も行っていた。これを示したのが図16（c）である。逆に文字

の削除などの操作によって、構造枠内の内容が減少した場合で、それに応じて構造枠を調整する場合も同様に行っていた。

【0088】このように構造枠内のレイアウト処理に続いて構造枠の再配置を行う場合には、枠の再配置計算とその結果の表示処理が頻繁に発生するため、ユーザが計算及び表示の処理待ちになるという欠点があった。

【0089】さらに、構造枠のサイズの変更に応じて編集集中の構造枠の位置が変更されるので、ユーザは、改めて編集集中の枠の位置と現在の挿入位置の確認を行わなければならない、ユーザの連続的な入力作業が妨げられるという欠点があった。

【0090】このような欠点を克服する実施形態例2について、図17および図18を参照して説明する。実施形態例2を用いると、構造枠内の編集を行っている最中に、構造枠サイズや位置の変更を行わなくてもすむ。

【0091】図17は本編集方式における構造枠サイズに対する内容あふれが発生した時点から構造枠のサイズ拡大と再配置までの一連の処理を示したものである。

【0092】まず、図17(a)は文字の入力の結果、現在挿入位置が枠内に留まらなくなった状態を表している。灰色で示されている文字は、最終行から送り出された後に配置される場所を仮想的に表したものである。このときの現在挿入位置から構造枠外に追い出されたことが判明する。

【0093】図17(b)は、追い出された内容を含め全体を表示するに足るサイズに内容領域の大きさを変更した様子を示している。

【0094】図17(c)は、現在挿入位置が枠内に表示される量だけ上方向にスクロールを行った様子を示している。本方式では、この時点では構造枠のサイズの変更や再配置は発生せず、現在挿入位置を構造枠内に表示しながら次の文字入力を待つことになる。

【0095】図17(d)は、ユーザが別の構造枠の編集開始を指示した様子を示しており、図17(e)は、この編集対象構造枠の変更を検出した時点で初めて構造枠のサイズを内容領域に合わせて拡大し、構造枠間の再配置を行った様子を示している。

【0096】図18は、実施形態例2の編集方式の動作手順を表すフローチャートである。

【0097】まず、新たな文字入力を判定し(ステップS601)、文字入力がないと判定された場合には再びS601を行う。

【0098】一方、文字入力が出検された場合には、その文字の入力により変更された現在挿入位置を検出し(ステップS602)、現在挿入位置が構造枠外であるかどうかを判定する(ステップS603)。ここで、もし構造枠内であると判定された場合には、再びステップS601に戻る。

【0099】一方、構造枠外であると判定された場合

は、続けて現在の内容領域外であるかどうかを判定する(ステップS604)。もし、内容枠外であると判定された場合は現在挿入位置を含め、全ての内容が表示可能なサイズに内容領域サイズを拡大する(ステップS605)。一方、既に存在する内容領域内である場合にはステップS605は行わない。

【0100】内容領域サイズが全内容を表示するに足る大きさになった時点で、今度は現在挿入位置が構造枠内に現れる量だけスクロールを行い(ステップS606)、枠内の再表示を行う(ステップS607)。

【0101】ここで、編集対象枠の変更指示が出検された場合には(ステップS608)、現在の内容領域サイズに合わせて構造枠のサイズの変更を行い(ステップS609)、最後に構造枠間の再配置を行って(ステップS610)、一連の処理が完了する。一方、対象枠の変更指示が出検されない場合には、ステップS601へ戻る。

【0102】ここでは、文字の入力に応じて逐一構造枠内の完全なレイアウトを行う必要はない。すなわち、ユーザが連続した入力操作を行っている場合には、現在の行間値を用いて一時的に構造枠外にはみ出すように入力を続行させ、入力操作が一定期間中断したことを検出して構造枠内の正式なレイアウトを行うこともできる。この様子を図19に示す。この場合でも、編集対象枠の変更がない限り枠のサイズ変更並びに再配置は起らないことは勿論である。

【0103】また、図17には枠内の任意の位置へとスクロールするためのスクローラが示されているが、この機能は本発明にとって必須の存在ではない。

【0104】文字入力検出ステップ(ステップS601)には、矢印キーなどによる現在挿入位置移動指示の検出も含めることができる。

【0105】さらに、図17(d)ではポインティングデバイスにより編集対象構造枠の変更が行われる例を示したが、これはキーボード上の例えば矢印キーや、メニューなどによって行われる場合もあり得る。

【0106】上述のような内容の増加が発生する場合だけでなく、文字削除などにより内容の減少が発生する場合の実施形態例2についても説明する。

【0107】ここでは図20ならびに図21を用いて、内容が減少する場合、実施形態例2がどのように動作するかを説明する。

【0108】図20は、文字削除の結果、構造内の全内容を表示するための内容領域の必要サイズが構造枠より小さくなった様子を示している。

【0109】図20(a)は、例えば文字がちょうど3行入る構造枠内で文字の削除が発生した結果2行になってしまった様子を示しており、図20(b)は図20(a)の状況で全内容を表示するのに必要な領域サイズを仮想的に示したものである。従来の方式の場合、行数

が2行になった段階で図20(b)に示した内容表示必要サイズに合わせて内容領域および構造枠のサイズを縮小し、さらに枠の再配置の処理を行うが、本方式においては別の構造枠への編集対象変更指示があるまでは、構造枠のサイズを変更しない。

【0110】図21に内容が減少する場合の実施形態例2の編集方式の動作手順を表すフローチャートを示した。

【0111】まず、文字削除の検出判定を行う(ステップS701)。文字削除が検出されない場合は再びステップS701に戻る。

【0112】一方、検出された場合には続けて現在挿入位置の検出を行う(ステップS702)。

【0113】次に、現在挿入位置が構造枠外であるかどうかの判定を行い(ステップS703)、もし構造枠外であれば、現在挿入位置が枠内に現れる量だけスクロールを行う(ステップS704)。一方、構造枠内に留まっている場合にはスクロール処理は行わない。

【0114】挿入位置が構造枠内であることが保証された段階で構造枠内の再表示を行う(ステップS705)。

【0115】さらに編集対象構造枠の変更指示があるかどうかを調べ(ステップS706)、変更指示がない場合にはステップS701へと戻る。一方、変更指示が検出された場合には、内容全体を表示するのに必要な領域サイズを求め(ステップS707)、そのサイズに合わせて構造枠のサイズを変更し(ステップS708)、構造枠の再配置を行う(ステップS709)。

【0116】本動作手順においては、ステップS706に示した編集対象の変更指示の検出時まで一切内容領域のサイズ変更を行っていないが、ステップS705の直前に内容領域サイズの計算処理を行い、構造枠サイズより大きい場合には構造枠サイズと同一サイズに変更することもできることは明らかである。

【0117】また、ステップS705における再表示の処理は必ずしも枠内全ての領域を行う必要はないというまでもない。

【0118】

【発明の効果】以上説明したように、本発明を用いることにより、独自の構造を使用することなく表形式レイアウトの実現が可能となった。このことにより、一般的な構造化エディタで作成・蓄積された文書を共有するような場合においても、表形式レイアウトで表示・印刷することが可能となった。

【0119】また、構造枠内の編集を行っている最中に、構造枠サイズや位置の変更を行わなくてもすむ。

【図面の簡単な説明】

【図1】構造化文書を説明する図である。

【図2】文書型定義を説明する図である。

【図3】構造化文書に対する体裁指定方式を説明する図である。

【図4】従来方式を用いた構造化エディタの表示例を説明する図である。

【図5】従来方式を用いた表形式表示を説明する図である。

【図6】本発明の一実施形態例の構造化文書編集システムの構成を示すブロック図である。

【図7】実施形態例1の表形式レイアウト方式を説明する図である。

【図8】実施形態例1の構造枠表示属性の例を説明する図である。

【図9】実施形態例1の構造枠の配置の考え方について説明する図である。

【図10】実施形態例1の構造枠配置指定の例を説明する図である。

【図11】実施形態例1の表形式のための配置指定の例を説明する図である。

【図12】実施形態例1の表形式表示を説明する図である。

【図13】実施形態例1の動作を説明するフローチャートである。

【図14】実施形態例1の動作を説明するフローチャートである。

【図15】実施形態例1の動作を説明するフローチャートである。

【図16】従来の構造枠編集を説明する図である。

【図17】実施形態例2の編集方式を説明する図である。

【図18】実施形態例2の編集の動作を説明するフローチャートである。

【図19】実施形態例2の他の例を説明する図である。

【図20】内容を減少する場合を説明する図である。

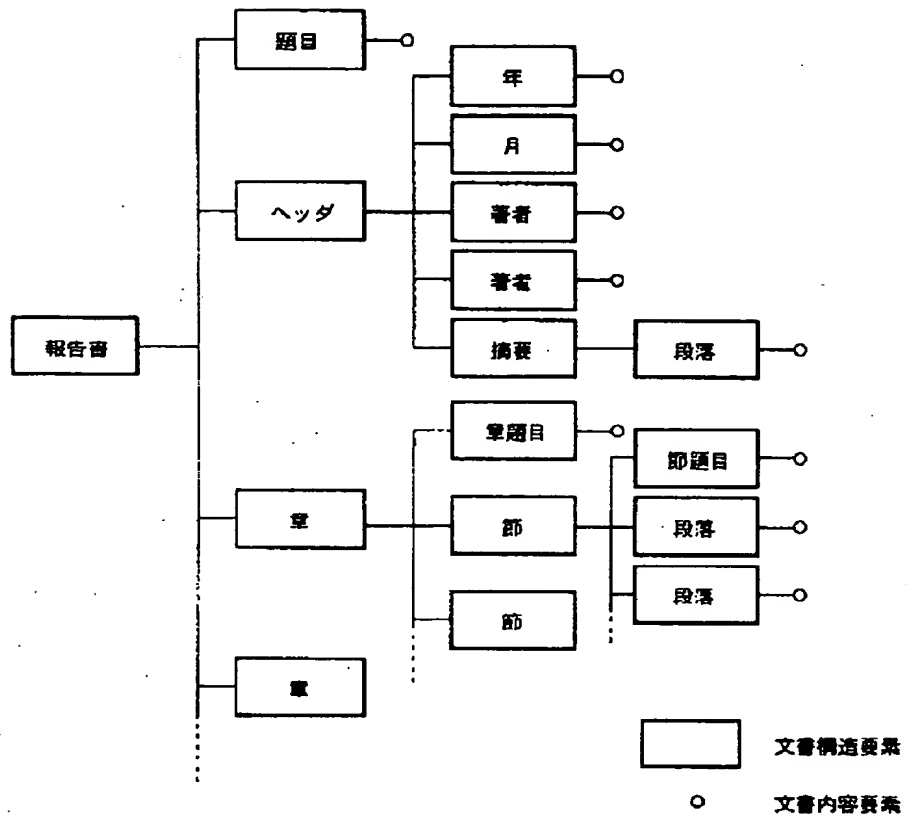
【図21】内容を減少する場合の実施形態例2の動作を説明するフローチャートである。

【符号の説明】

- 1 CRT
- 2 VRAM
- 3 BMU
- 4 KBD
- 5 PD
- 6 CPU
- 7 ROM
- 8 RAM
- 9 HDD
- 10 FDD
- 11 Net-I/F
- 12 I/O BUS

【図1】

(a)



(b)

<報告書>
 <題目>○○○の××× </題目>
 <ヘッダ>
 <年> 1994 </年> <月> 06 </月>
 <著者> ○山×大 </著者>
 <著者> ○川☆子 </著者>
 <摘要>
 <段落>
 ○○を△△することを試みた。その結果、.....

 </段落>
 <摘要>
 <ヘッダ>
 <章> <章題目> △△△ </章題目>
 <節> <節題目> ▽▽▽ </節題目>
 <段落>
 ××××××××××

 </段落>

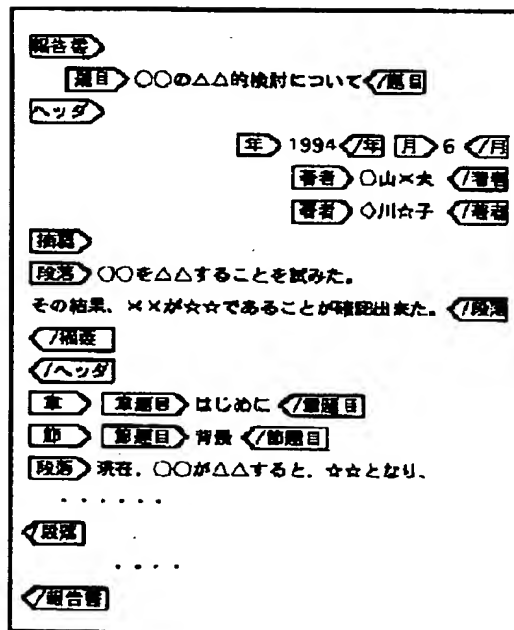
 </報告書>

【図2】

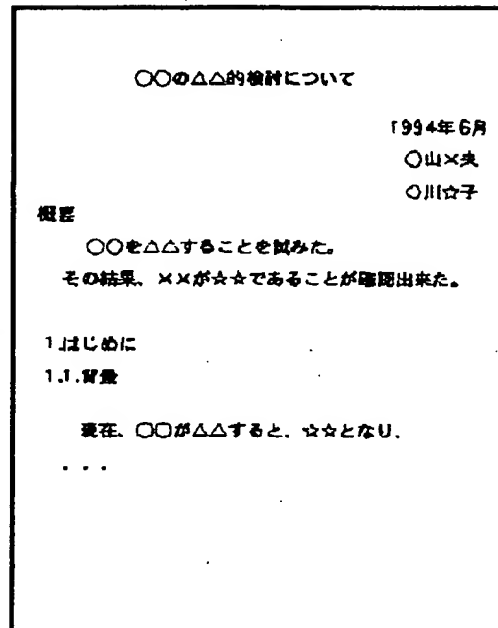
<!ELEMENT	報告書	--	(題目、ヘッダ、(章)*)	>	(.段落		
<!ELEMENT	題目	--	(#PCDATA)	>	(FontFace	"明朝"	; 明朝フォント
<!ELEMENT	ヘッダ	--	(年、月、(著者)+、概要)	>	(FontStyle	"標準"	; 標準スタイル
<!ELEMENT	年	--	(#PCDATA)	>	(FontSize	12)	; 12ポイント
<!ELEMENT	月	--	(#PCDATA)	>	(HeadBreak	ON)	; 開始行改行
<!ELEMENT	著者	--	(#PCDATA)	>	(DefaultIndent	12)	; 開始行字下げ(12ポイント)
<!ELEMENT	概要	--	(段落)*	>)		
<!ELEMENT	段落	--	(#PCDATA)	>			
<!ELEMENT	章	--	(章題目、(節)*)	>	(.章題目		
<!ELEMENT	章題目	--	(#PCDATA)	>	-----		
<!ELEMENT	節	--	(節題目、(段落)*)	>)		
<!ELEMENT	節題目	--	(#PCDATA)	>			

【図3】

【図4】

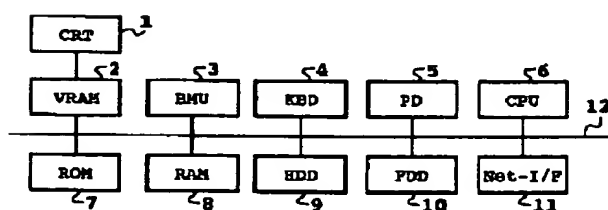


(a)

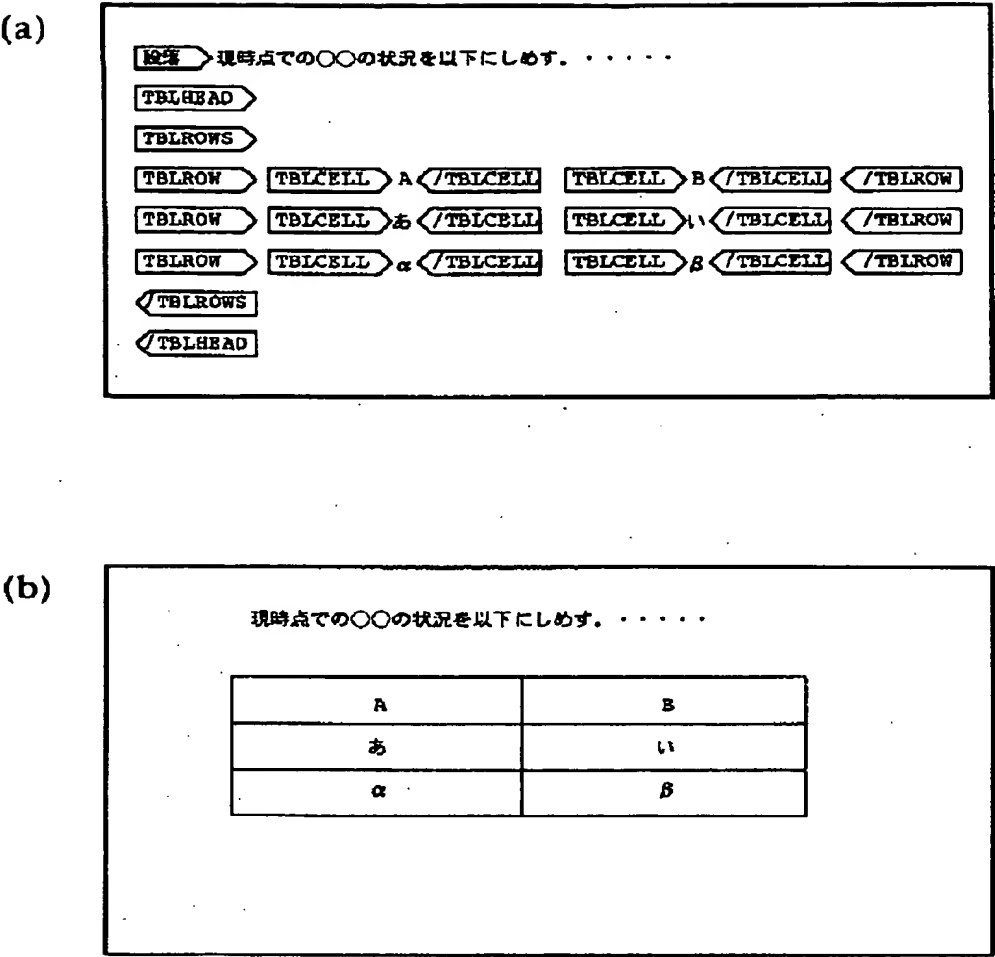


(b)

【図6】



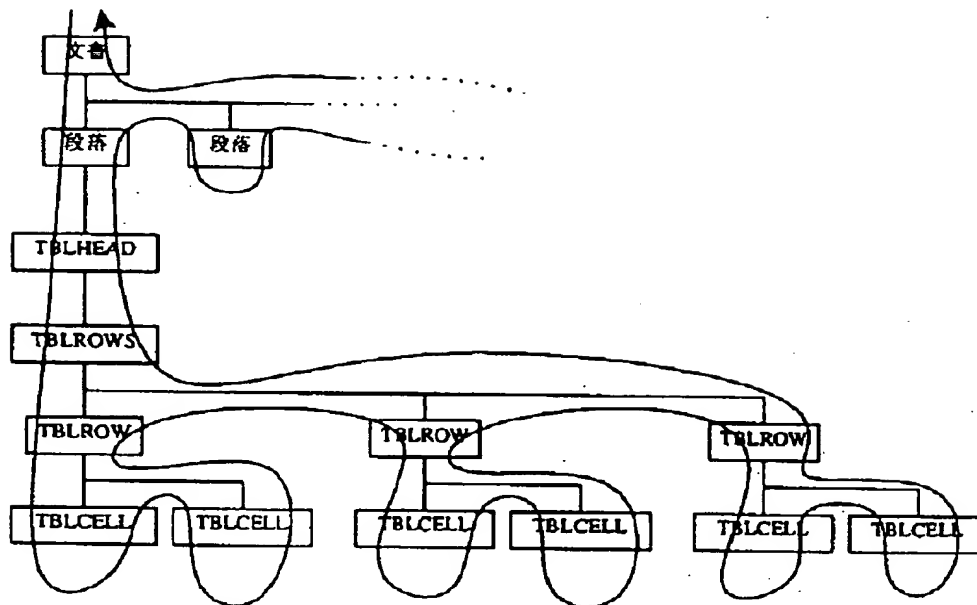
【図5】



【図8】

論理構造名	構造枠表示の性質	構造枠表示時に構造名を表示するか?
文書	構造枠表示指示に従う	構造名を表示する
段落	構造枠表示指示に従う	構造名を表示する
TBLHEAD	必ず構造枠を表示	構造名を表示しない
TBLROWS	必ず構造枠を表示	構造名を表示しない
TBLROW	必ず構造枠を表示	構造名を表示しない
TBLCELL	必ず構造枠を表示	構造名を表示しない

【図7】



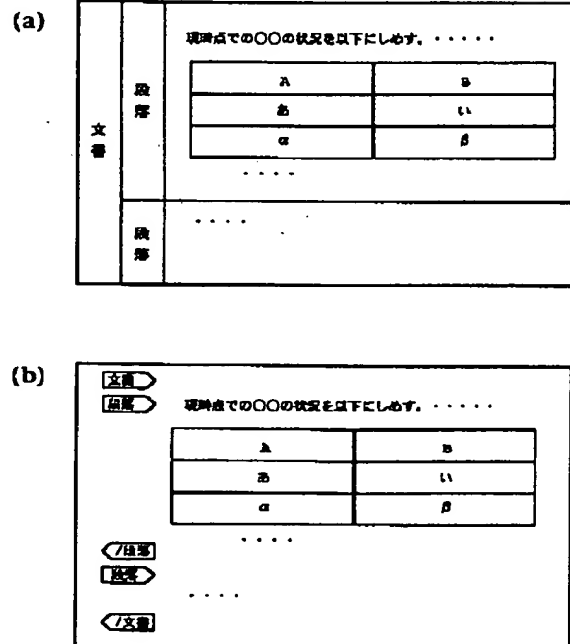
【図10】

```

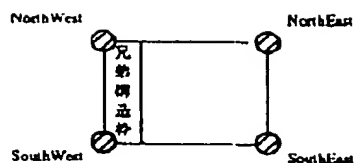
1  (TBLHEAD
2  (TBLROWS
3      (Attachment      Parent)
4      (AttachOrigin    NorthWest)
5      (XOffset          0)
6      (YOffset          0)
7  )
8  )
9
10 (TBLROWS
11 (TBLROW
12     (Attachment      TBLROW)
13     (AttachOrigin    SouthWest)
14     (XOffset          0)
15     (YOffset          0)
16 )
17 )
18
19 (TBLROW
20 (TBLCELL
21     (Attachment      TBLCELL)
22     (AttachOrigin    NorthEast)
23     (XOffset          0)
24     (YOffset          0)
25 )
26 )

```

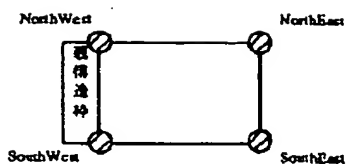
【図12】



【図9】

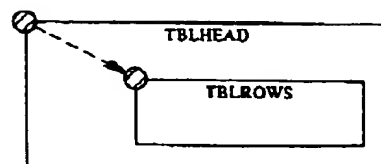


(a)

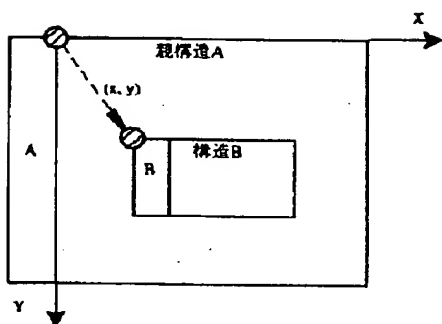


(b)

【図11】

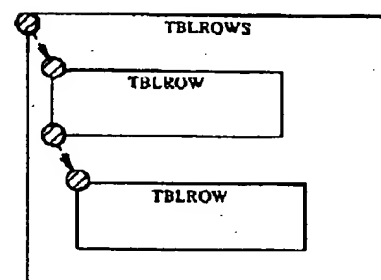


(a)

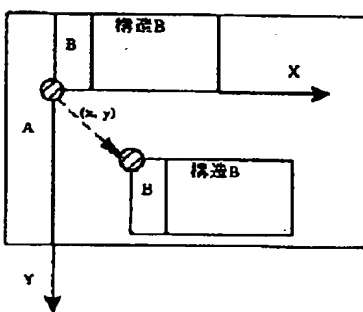


(c)

```
(A
  (B
    (Attachment      Parent)
    (AttachOrigin    NorthWest)
    (XOffset         x)
    (YOffset         y)
  )
)
```

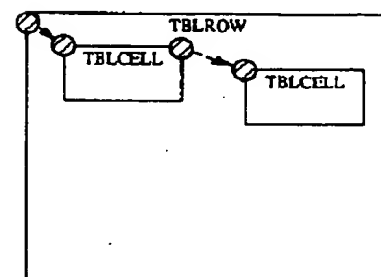


(b)



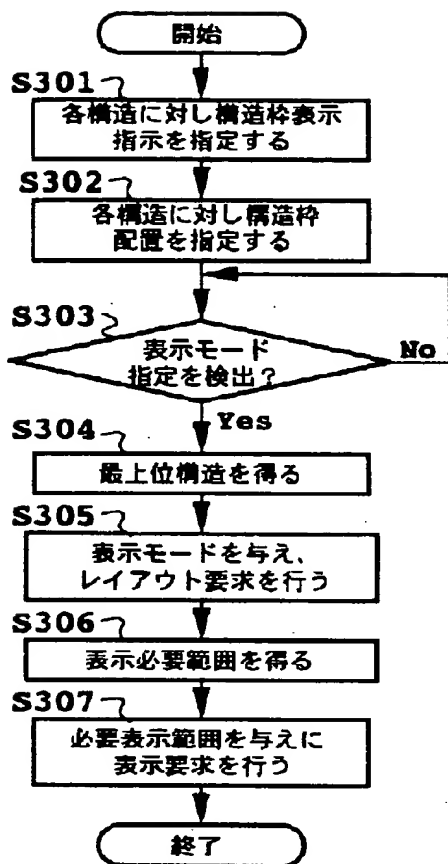
(d)

```
(A
  (B
    (Attachment      B)
    (AttachOrigin    SouthWest)
    (XOffset         x)
    (YOffset         y)
  )
)
```

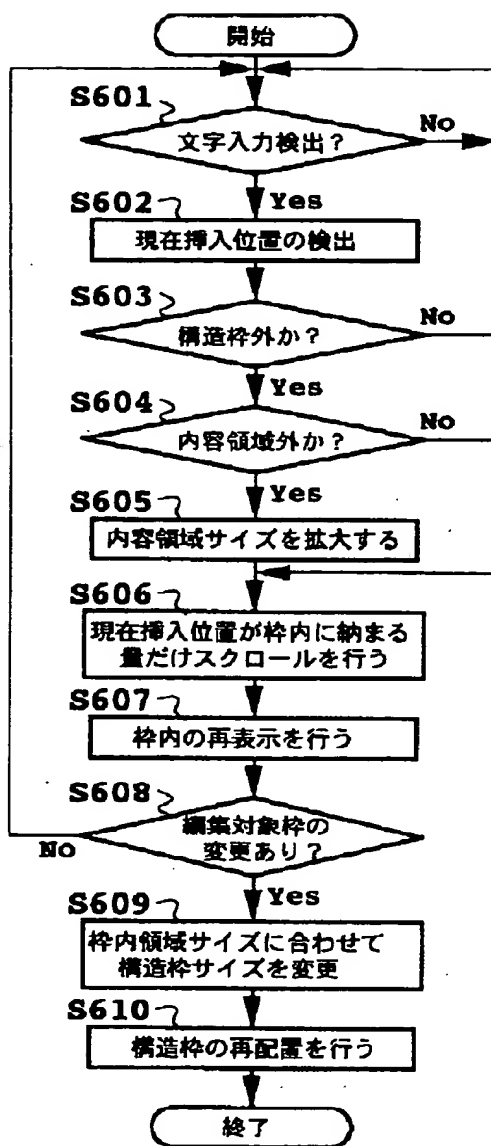


(c)

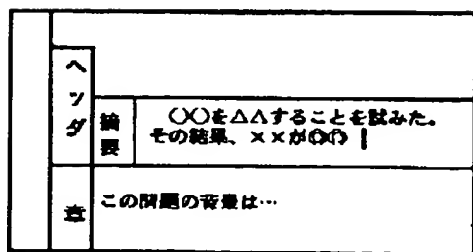
【図13】



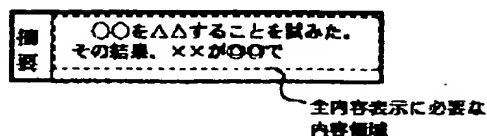
【図18】



【図20】

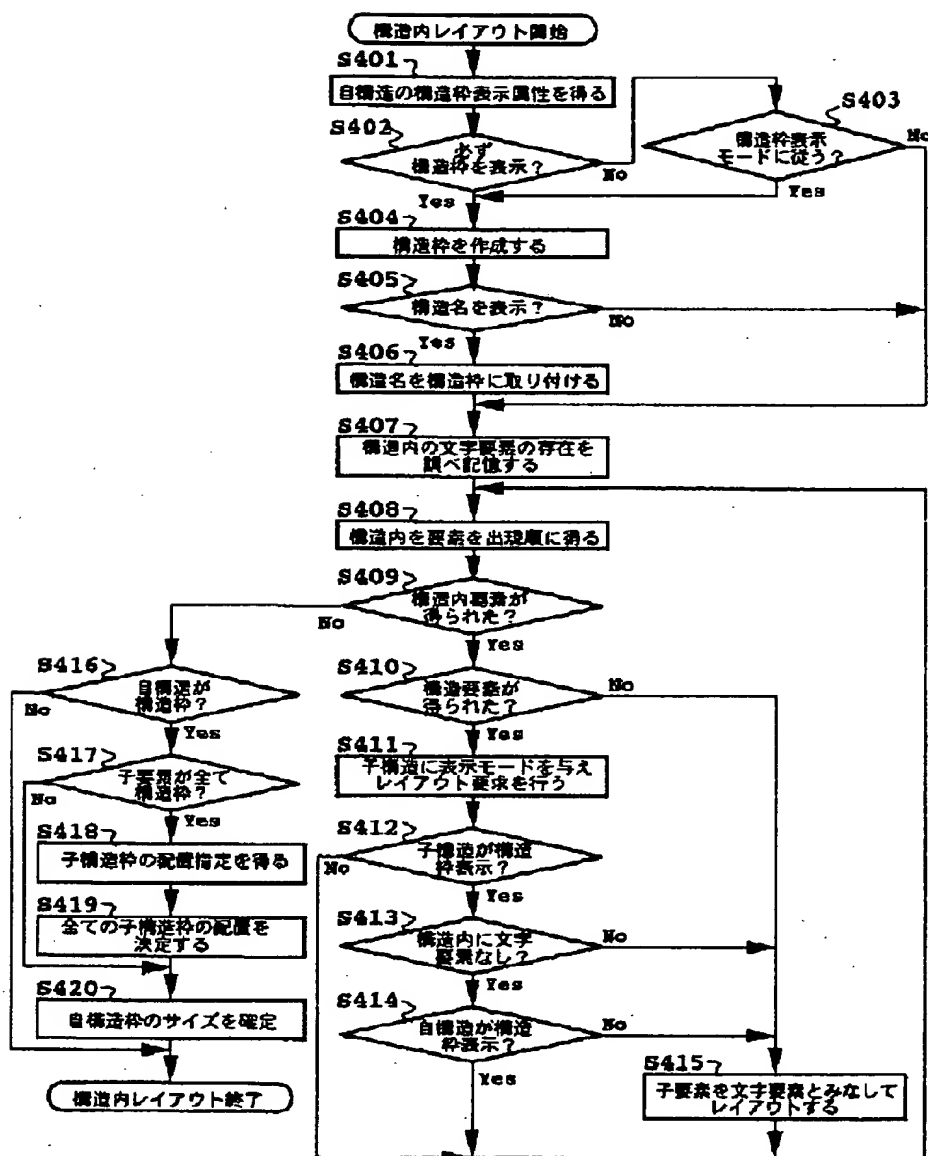


(a)

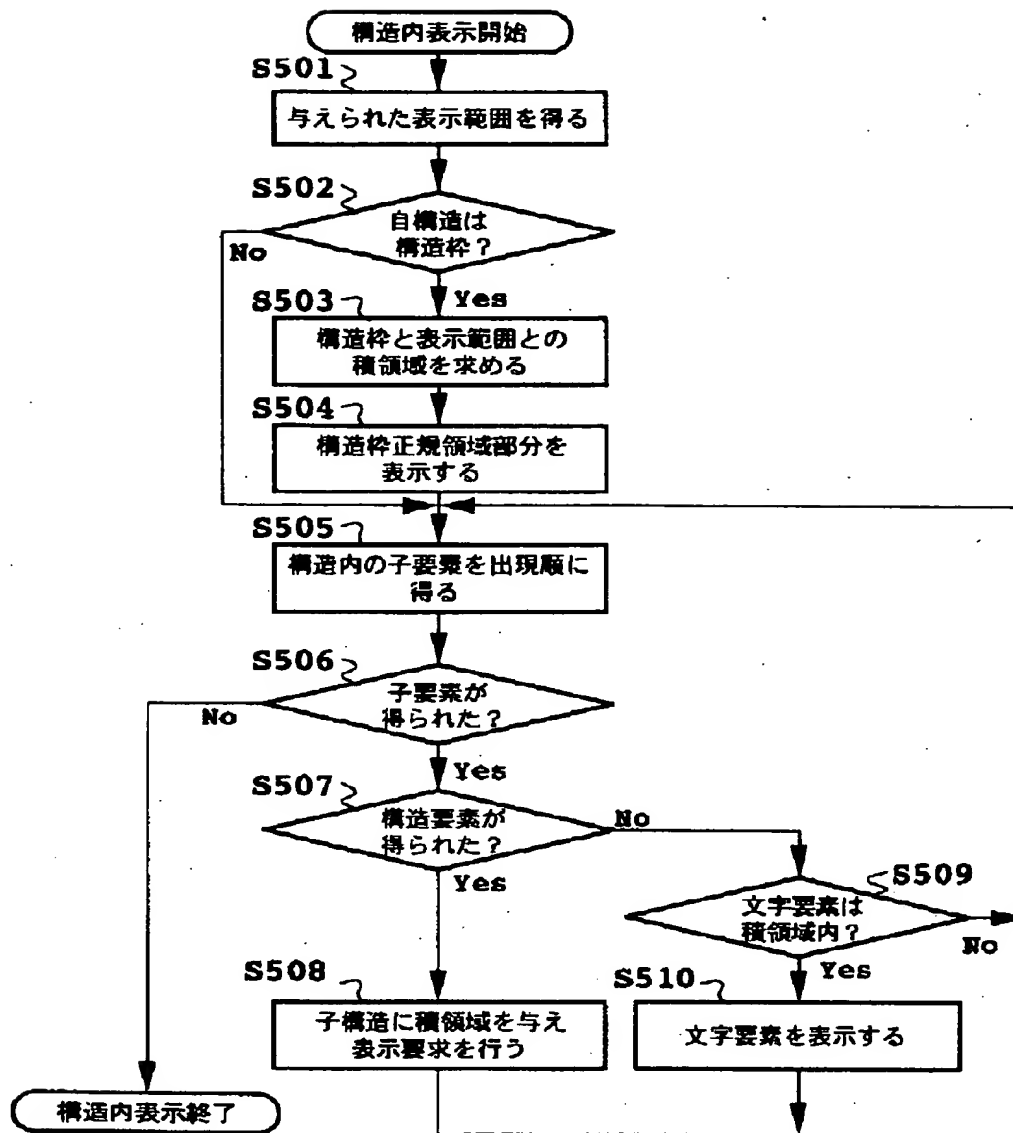


(b)

【図14】



【図15】



【図16】

報告書	題目	〇〇の××的検討について
	年	1993
	月	12
	著者	〇山×夫
	著者	◇川×子
	著者	
	摘要	〇〇を△△することを試みた。 その結果、××が〇〇であるという 問題が発生し、早急に解決策を講じ る必要が
章	この問題の背景は…	

(a)

報告書	題目	〇〇の××的検討について
	年	1993
	月	12
	著者	〇山×夫
	著者	◇川×子
	著者	
	摘要	〇〇を△△することを試みた。 その結果、××が〇〇であるという 問題が発生し、早急に解決策を講じ る必要が
章	この問題の背景は…	

(b)

報告書	題目	〇〇の××的検討について
	年	1993
	月	12
	著者	〇山×夫
	著者	◇川×子
	著者	
	摘要	〇〇を△△することを試みた。 その結果、××が〇〇であるという 問題が発生し、早急に解決策を講じ る必要が
章	この問題の背景は…	

(c)

【図17】

ヘッダ		
	概要	〇〇を△△することを試みた。 その結果、××が〇〇であるという 問題が発生し、早急に解決策を講じ る。
章	この問題の背景は…	

(a)

ヘッダ		
	概要	〇〇を△△することを試みた。 その結果、××が〇〇であるという 問題が発生し、早急に解決策を講じ る。
章	この問題の背景は…	

(b)

ヘッダ		
	概要	〇〇を△△することを試みた。 その結果、××が〇〇であるという 問題が発生し、早急に解決策を講じ る。
章	この問題の背景は…	

(c)

ヘッダ		
	概要	〇〇を△△することを試みた。 その結果、××が〇〇であるという 問題が発生し、早急に解決策を講じ る。
章	この問題の背景は…	

(d)

ヘッダ		
	概要	〇〇を△△することを試みた。 その結果、××が〇〇であるという 問題が発生し、早急に解決策を講じ る。
章	この問題の背景は…	

(e)

【図19】

報告書	題目	(X)のX X的検討について
	年	1993
	月	12
	著者	○山X夫
	著者	◇川☆子
	著者	
	摘要	○○も△△することを試みた。その結果、XXが○○であるという問題が発生し、早急に解決策を講じる必要が発生したので、○○のX X的検討を行い、いくつかの対策を
章		

【図21】

